NEW UTILITY PATENT APPLICATION TRANSMITTAL

(Only for new nonprovisional applications under 37 C.F.R. 1.53(b))

Docket No.
M4065.0290/P290
Total pages in this submission
52

TO THE COMMISSIONER FOR PATENTS
Box Patent Application
Washington, D.C. 20231

Transmitted herewith for filing under 35 U.S.C. 111(a) and 37 C.F.R. 1.53(b) is a new utility patent application for an invention entitled:

> ### DRAM WITH HIDDEN REFRESH

and invented by:

> ### Kevin J. Ryan

IF A CONTINUATION APPLICATION, check appropriate box and supply requisite information:

☐ Continuation   ☐ Divisional

☐ Continuation-in-part (CIP) of prior application No.: _____

Enclosed are:

Application Elements

1. ☐ Filing fee as calculated and transmitted as described below

2. ☒ Specification having _____30_____ pages(s) and including the following:

    a. ☒ Descriptive title of the invention

    b. ☐ Cross references to related applications *(if applicable)*

    c. ☐ Statement regarding Federally-sponsored research/development *(if applicable)*

    d. ☐ Reference to microfiche appendix *(if applicable)*

    e. ☒ Background of the invention

    f. ☒ Brief summary of the invention

    g. ☒ Brief description of the drawings *(if drawings filed)*

    h. ☒ Detailed description

    i. ☒ Claims as classified below

    j. ☒ Abstract of the disclosure

Application Elements (continued)

3. [X] Drawing(s) *(when necessary as prescribed by 35 U.S.C. 113)*

    [ ] Formal   [X] Informal   Number of sheets: 7

4. [X] Oath or Declaration

    a. [X] Newly executed (original or copy)   [ ] Unexecuted

    b. [ ] Copy from a prior application (37 C.F.R. 1.63(d)) *(for continuation/divisional applications only)*

    c. [ ] With Power of Attorney   [X] Without Power of Attorney

5. [ ] Incorporation by reference *(usable if Box 4b is checked)*
The entire disclosure of the prior application, from which a copy of the oath or declaration is Supplied under Box 4b, is considered as being part of the disclosure of the accompanying application and is hereby incorporated by reference therein.

6. [ ] Computer program in microfiche

7. [ ] Genetic sequence submission *(if applicable, all must be included)*

    a. [ ] Paper copy

    b. [ ] Computer readable copy

    c. [ ] Statement verifying identical paper and computer readable copies

Accompanying Application

8. [X] Assignment papers *(cover sheet & document(s))*

9. [X] Power of Attorney and 37 C.F.R. 3.73(b) Statement *(when there is an assignee)*

10. [ ] English translation document *(if applicable)*

11. [ ] Information Disclosure Statement/PTO-1449   [ ] Copies of IDS citations

12. [ ] Preliminary Amendment

13. [X] Acknowledgment postcard

14. [ ] Certified copy of priority document(s) *(if foreign priority is claimed)*

15. [ ] Certificate of Mailing

    [ ] First Class   [ ] Express Mail (Label No.: _____)

16. [ ] Small Entity statement(s) -- # submitted _____ *(if Small Entity status claimed)*

Accompanying Application (continued)

17. ☐ Additional enclosures *(please identify below):*

Fee Calculation and Transmittal

The filing fee for this utility patent application is calculated as follows:

☒ Large Entity   ☐ Small Entity

CLAIMS AS FILED					
For	**# Filed**	**# Allowed**	**# Extra**	**Rate**	**Fee**
Total Claims	54	- 20 =	34	x $18.00	$ 612.00
Independent Claims	5	- 3 =	2	x $78.00	156.00
Multiple Dependent Claims (check if applicable)			☐		
Other Fees (specify purpose): Recording fee					40.00
				BASIC FEE	690.00
				TOTAL FILING FEE	$1498.00

☒ A check in the amount of $1498.00 to cover the total filing fee is enclosed.

☒ The Commissioner is hereby authorized to charge Deposit Account No. 4 - 1073 as described below. A duplicate copy of this sheet is enclosed.

☐ Charge the amount of _____ as filing fee.

☒ Credit any overpayment.

☒ Charge any additional filing fees required under 37 C.F.R. 1.16 and 1.17.

☐ Charge the issue fee set in 37 C.F.R. 1.18 at the mailing of the Notice of Allowance, pursuant to 37 C.F.R. 1.31(b).

Please direct all correspondence to:

Thomas J. D'Amico
Dickstein Shapiro Morin & Oshinsky LLP
2101 L Street NW
Washington, DC 20037-1526

Telephone (202) 785-9700
Facsimile   (202) 887-0689

_____   Dated: August 21, 2000
Thomas J. D'Amico

Attorney Reg. No. 28,371
Dickstein Shapiro Morin & Oshinsky LLP
2101 L Street NW
Washington, DC 20037-1526
(202) 785-9700

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION FOR U.S. LETTERS PATENT

Title:

DRAM WITH HIDDEN REFRESH

Inventor:

Kevin J. Ryan

Dickstein Shapiro Morin &
Oshinsky LLP
2101 L Street NW
Washington, DC 20037-1526
(202) 785-9700

DRAM WITH HIDDEN REFRESH

BACKGROUND OF THE INVENTION

5   ### 1. FIELD OF THE INVENTION

The present invention relates to the field of integrated circuits. More

particularly, it relates to a dynamic random access memory (DRAM) refresh

operation that does not depend upon a certain number of read accesses and that

also does not interfere with data accesses initiated by the controller.

10

2. DESCRIPTION OF THE RELATED ART

In a dynamic random access memory ("DRAM"), data is stored as a

logic one or zero by the presence or absence of charge on a capacitor within an

individual memory cell. After the data has been stored as charge on the

15   capacitor, the charge gradually leaks off and the data is corrupted. The time

within which a refresh must be performed lest the data might be in danger of

being lost is commonly referred to as the refresh interval. Therefore, a "refresh"

cycle must be performed to maintain the integrity of the data. To refresh data in

a memory array, the array is typically placed in a read mode to obtain the present

20   data stored in a row of memory cells. Subsequently, this data is used as new

input data that is re-written into the row of memory cells, thus maintaining the

stored data. An important aspect of the refresh cycle of prior art DRAMs is that

no other operation involving a different row in the array can occur simultaneously during the refresh operation.

A functional block diagram of a typical 64M SDRAM is shown in FIG.
5   1. Since SDRAM operation is well known in the art, only a brief description of the Fig. 1 DRAM will be provided herein. The Fig. 1 SDRAM is a quad-bank ("x16") SDRAM having four 16,777,216-bit banks 10A-10D organized as 4,096 rows by 256 columns by 16 bits. Circuit blocks 12A-12D include sense amplifiers coupled to each column within the memory array to transform charge
10   on the capacitor in the memory cell into a valid logic one or zero. Read and write accesses to the SDRAM are burst oriented; accesses start at a selected location and continue for a programmed number of locations in a programmed sequence. Address bits registered coincident with an ACTIVE command are used to select the bank and row to be accessed. The address bits registered
15   coincident with the READ or WRITE command are used to select the starting column location for the burst access.

In the DRAM of FIG. 1, two types of refresh cycles are available; auto refresh and self-refresh. The self-refresh cycle automatically and internally
20   refreshes the data sequentially in the memory arrays. The auto refresh cycle is analogous to CAS#-BEFORE-RAS# (CBR) REFRESH in conventional DRAMs. Providing a distributed AUTO REFRESH command every 15.625µs will meet the refresh requirement and ensure that each row is refreshed. The SELF

REFRESH command can be used to retain data in the SDRAM even if the rest of the system is powered down. Once self refresh mode is engaged, the SDRAM provides its own internal clocking, causing it to perform its own AUTO REFRESH cycles.

5

One problem associated with such refresh cycles is that a given refresh cycle may conflict with a controller-generated access command (e.g., a read or write command). Many DRAM's are configured for deterministic latency which means that an access can never lose priority; and therefore, the refresh operation

10 must wait (e.g., put into a queue) until the access has been completed. The danger with such a practice, of course, is that a refresh operation may be postponed for a period of time greater than a predetermined refresh interval, thus placing the data at risk.

15 One approach to solving this problem has been proposed by Monolithic System Technology, Inc. with its 1T-SRAM technology. Under the 1T-SRAM approach, refresh operations are triggered by read commands. One problem with relying upon read commands to trigger a refresh operation is that it can result in refresh overkill. That is, a read command may be received from

20 the system processor more frequently than a refresh is actually required, thus, wasting valuable power resources. Another problem with relying upon read commands is that if the memory is idle for greater than the refresh interval (e.g., 64ms), data will be lost.

Yet another approach to the problem proposes to trigger a refresh operation off of a clock pulse (e.g., after a predetermined number of clock pulses, a refresh operation is triggered). However, under this approach, there is no way

5   to guarantee that the refresh will not begin just before a read command. In such a case, if the read access is delayed until the refresh operation is completed, the overall time required to access memory is increased.

Another approach to the refresh problem is disclosed in U.S. Patent

10   No. 6,028,804 (the "'804 patent"). The '804 patent discloses a method of operating a memory array which contains memory cells requiring periodic refresh in which a refresh is performed only if no external access is determined to be pending. An accumulator is disclosed for accumulating (i.e., delaying) refresh requests that conflict with an external access for up to seven refresh requests (or

15   56μs).

According to the '804 patent, external memory accesses are allowed to continue for a period of up to 56μs without losing refresh cycles. The '804 patent states that back-to-back external accesses longer than 56μs generally do not occur. The '804 patent also states that the memory cycle of its disclosed

20   memory system is equal to one clock cycle. Thus, the '804 patent does not consider the special complex problems associated with having a memory cycle that is equal to multiple clock cycles, such as e.g., the fact that access commands must be disallowed for certain clock cycles. In addition, the '804 patent does

not provide for simultaneous operations (e.g., read, write or refresh). Thus, there exists a need for a hidden DRAM refresh operation that does not depend upon a certain number of read accesses, that does not interfere with data accesses initiated by the controller, that allows for simultaneous operations and that is

5   consistent with today's memory cycles which may last for multiple clock cycles.

SUMMARY OF THE INVENTION

The present invention overcomes the problems associated with the prior art and provides a system and method for performing a hidden DRAM

10   refresh operation that does not depend upon a certain number of read accesses and that also does not interfere with data accesses initiated by the controller. In accordance with an exemplary embodiment of the invention, a synchronous DRAM is provided having specified time slots (e.g., every multiple of 4 clock pulses) for entering read or write commands. In addition, a self-refresh counter

15   and controller are provided and which are driven by the DRAM input clock. During operation, the DRAM performs internally generated refresh operations on a periodic basis while avoiding collisions with controller-generated data accesses. Specifically, the command and address bus contains predetermined time slots (e.g., every fourth positive edge of the DRAM input clock after the

20   first read or write command is entered) within which the controller is allowed to apply data access commands (e.g., read, write). An internal refresh cycle can be executed without interfering with any data accesses by starting the refresh after decoding a non-conflicting command in one of these time slots and finishing

before the next command time slot. If an internal refresh operation is delayed

(e.g., by the decoding of a conflicting access command) it will be completed at

the earliest opportunity thereafter. In accordance with an exemplary

embodiment of the invention, NOP's (no operation's), accesses to other memory

5   arrays (also referred to as banks), and accesses to an accompanying static random

access memory (SRAM) cache are all examples of non-conflicting commands and

all provide such an opportunity to complete the refresh operation. In worst case

scenarios, it is the SRAM cache that ensures that an opportunity to refresh will

occur within the allotted refresh interval.

10

BRIEF DESCRIPTION OF THE DRAWINGS

The foregoing and other advantages and features of the invention will

become more apparent from the detailed description of preferred embodiments

of the invention given below with reference to the accompanying drawings in

15   which:

Fig. 1 illustrates a block diagram of an exemplary prior art DRAM;

Fig. 2 illustrates a simplified block diagram of a DRAM refresh system

in accordance with an exemplary embodiment of the invention;

Fig. 3 illustrates a flowchart depicting a functional flow of the Fig. 2

20   system, in accordance with an exemplary embodiment of the invention;

Fig. 4 illustrates a READ – READ timing diagram for the Fig. 2

system, in accordance with an exemplary embodiment of the invention;

Fig. 5 illustrates a WRITE – WRITE timing diagram for the Fig. 2 system in accordance with an exemplary embodiment of the invention;

Fig. 6 illustrates a READ – WRITE – READ timing diagram for the Fig. 2 system, in accordance with an exemplary embodiment of the invention;

5   and

Fig. 7 illustrates a block diagram of a processor-based system incorporating a DRAM refresh system in accordance with an exemplary embodiment of the invention.

10   <u>DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS</u>

The present invention will be described as set forth in an exemplary embodiment illustrated in Figs. 2-7. Other embodiments may be utilized and structural or logical changes may be made without departing from the spirit or scope of the present invention.

15

Referring to Fig. 2, a block diagram of a DRAM having a hidden refresh system is depicted in accordance with an exemplary embodiment of the invention. The structure of the Fig. 2 DRAM is essentially the same as that of the Fig. 1 DRAM except that the Fig. 2 DRAM contains a SRAM cache 240, a

20   cache controller 250 and a refresh controller 230. The address bus 260 is coupled to the cache controller 250 for receiving all 14 bits of the address. The cache controller 250 is coupled to the SRAM cache 240 for controlling the SRAM cache 240. An input of the SRAM cache 240 is coupled to bus 255 and

an output of the SRAM cache 240 is coupled to bus 275. The operation of the Fig. 2 diagram will be described in connection with the flowchart of Fig. 3.

Turning to Fig. 3, a flowchart is illustrated and depicts an exemplary
5   functional flow of the system described above in connection with Fig. 2. The operation begins at step S300 and at step S302, the refresh controller 230 determines whether an initial Read or Write command has been received over the command/address bus 260. If not, step S302 is repeated until an initial command has been received. At step S304, the self-refresh counter 210 counts
10   positive clock edges of the DRAM input clock 100 which, for example, may be running at 300 MHz.

At step S306, the refresh controller 230 determines whether four clock pulses have been counted since the initial command has been received. If
15   not, step S306 is repeated. At step S308, the refresh controller 230 determines whether it's time to perform a refresh operation on a given row of memory cells with a particular memory array 10A-10D.

For example, assuming a 64M synchronous DRAM with 4 internal
20   banks of 4096 rows each, and for each refresh operation, one row in each bank is refreshed. That is, there are 16,384 total rows in the device and 4 rows are refreshed at a time, meaning 4096 refresh operations are needed to refresh all memory cells. Each memory cell must be refreshed at least once every 64ms,

therefore, one refresh operation should be performed every 15.625μs (i.e., 4096

x 15.625μs = 64ms). The self-refresh counter 210 counts the number of input

clock pulses corresponding to 15.625μs and generates an internal interrupt

indicating that it is time to perform a refresh operation. It is also possible to

5   simultaneously refresh one row in each of a plurality of sub-arrays in a bank

containing sub-arrays, while leaving the other banks available for memory access,

and cycle the refresh operation among all available memory banks.

          If the refresh controller 230 has determined it is not yet time to

10   perform a refresh, step S306 is repeated. If it is determined that it is time to

perform a refresh operation, the refresh controller 230 determines whether the

DRAM has received a data access command from the system processor at step

S310. If not, the refresh operation is performed at step S312 and the process

returns to the input of portion S306. The actual refreshing of the memory cells

15   may be carried out in any number of ways known in the art including, but not

limited to, those methods described in U.S. Patent No. Re. 36,180 assigned to

Micron Technology, the contents of which are fully incorporated herein by

reference. Therefore, a specific process for actually carrying out the refresh

operation in connection with the present invention will not be described herein.

20

          If at step S310, the refresh controller 230 determines that a data

access command has been received from the system processor, the refresh

controller 230 determines whether the access command does not conflict with

the refresh operation at step S314. The refresh controller 230 is configured to continually run a predetermined refresh address sequence, the exact configuration of which is not critical for purposes of explaining the present invention. The present invention determines, among other things, whether the

5   refresh operation, scheduled to be performed on a particular row at a predetermined time, will conflict with a data access command received on the command/address bus 260.

If at step S314, it is determined that the data access command is non-

10 conflicting with the refresh operation, then upon decoding the non-conflicting data access command, the refresh controller 230 sees that the refresh operation is carried out on the particular row at portion S316 and the process returns to the input of portion S306. In accordance with a preferred embodiment of the invention, the refresh operation is completed between the time the non-

15 conflicting data access command is decoded and the next four clock pulses are counted from the DRAM input clock 220 (i.e., the refresh operation is completed before the next data access command is allowed to be received on the command/address bus 260).

20 If at step S314 it is determined that the data access command is not non-conflicting (i.e., is conflicting) with a scheduled refresh operation, then, at step S318, the refresh controller 230 is configured to wait for the conflicting data access command to be completed and postpones (i.e., queues) the refresh

operation until a next available opportunity, as will be explained more fully below.

As depicted in the Fig. 3 flowchart, the method is a continuous loop. The SRAM cache 240 is configured such that the refresh interval would never be violated. The cache 240 size equals the size of one sub-array (e.g., 10A of Fig. 2; however, it should be noted that one sub-array may or may not equate to one logical bank) and the refresh interval is set such that the time required to fill the cache 240 with the data from the sub-array 10A is less than the refresh interval. Therefore, even for the worst case scenario where every portion of the sub-array 10A was accessed once before any one portion was accessed a second time (i.e., where such an access would actually perform the refresh operation) a second refresh request would still not have arrived yet. That is, the first queued up request would be executed.

In accordance with an exemplary embodiment of the invention, the data included within the row to be accessed is written into the SRAM cache 240 where it may be accessed by the system processor for a desired read or write operation. In the meantime, data in other rows of the same sub-array is refreshed before the refresh interval is reached.

For example, a worst case scenario would be a continued series of read/write commands applied on every 4th clock pulse for a long time (i.e.,

several times greater than the refresh interval), whereby the read/write

commands are continually received by the same bank. Assuming there are e.g.,

four banks (e.g., 10A-10D), then during the continuing accesses to the same

bank (e.g., 10A), all the refresh operations could be directed to the other three

5   banks (e.g., 10A-10C) and eventually the data in the bank being accessed will be

lost (i.e., the stored charge will leak off the capacitors). It should be noted,

however, that the act of accessing rows in the one bank (e.g., 10A) will refresh

those rows, so the problem is when one bank is continually accessed, but not all

rows within the bank. In such a case, the non-accessed rows in the bank being

10   continually accessed (10A) will eventually be lost. This is where the SRAM cache

240 is effective. Accessing one bank continually for a long period of time means

that you must access some row/column combination in that bank multiple

times. With the SRAM cache 240, in accordance with the exemplary

embodiment of the invention, those multiple accesses will be serviced by the

15   SRAM cache 240, thereby freeing up the non-accessed rows in the continually

accessed bank (10A) in order to perform a much needed refresh.

Turning to Fig. 4, a Read – Read timing diagram is depicted in

accordance with the exemplary embodiment of the invention. Thirteen clock

20   pulses of a DRAM input clock are depicted as T_0-T_{12}. At each multiple of four

clock pulses, a read and/or write command is allowed to be placed on the

command/address bus 260 (of Fig. 2). Fig. 4 depicts a read command located

at clock pulses T_0, T_4 and T_8. In accordance with the exemplary embodiment of

the invention, the refresh operation may be performed between T0 and T4 and also between T4 and T8 and so on.

Turning to Fig. 5, a Write – Write timing diagram is depicted in

5   accordance with the exemplary embodiment of the invention. Thirteen clock pulses of a DRAM input clock are depicted as T_0-T_{12}. At each multiple of four clock pulses, a read and/or write command is allowed to be placed on the command/address bus 260 (of Fig. 2). Fig. 5 depicts a write command located at clock pulses T_0, T_4 and T_8. In accordance with the exemplary embodiment of

10   the invention, the refresh operation may be performed between T0 and T4 and also between T4 and T8 and so on.

Fig. 6 depicts a Read – Write – Read timing diagram, in accordance with the exemplary embodiment of the invention. Seventeen clock pulses of a

15   DRAM input clock are depicted as T_0-T_{16}. At each multiple of four clock pulses, a read and/or write command is allowed to be placed on the command/address bus 260 (of Fig. 2). Fig. 6 depicts a read command located at clock pulses T_0 and T_{12}. In addition, Fig. 6 depicts a write command located at clock pulse T_8. In accordance with the exemplary embodiment of the invention, the refresh

20   operation may be performed between T_0 and T_4 and also between T_4 and T_8 and so on. It should be noted that T_4 contains a NOP which can trigger a refresh operation if one is needed at that time; and, of course, the NOP will not conflict with the refresh operation.

Fig. 7 illustrates a block diagram of a processor-based system 700

utilizing a DRAM 308 having a refresh system constructed in accordance with

the present invention. That is, the DRAM 708 may be the DRAM illustrated in

5   Fig. 2 and operates as described above with respect to Figs. 3 to 6. The

processor-based system 700 may be a computer system or any other system

requiring a DRAM. The system 700 includes a central processing unit (CPU)

702, e.g., a microprocessor, that communicates with the DRAM 708 over a bus

720. It must be noted that the bus 720 may be a series of buses and bridges

10   commonly used in a processor-based system, but for convenience purposes only,

the bus 720 has been illustrated as a single bus. An input/output (I/O) device

706 may also be connected to the bus 720, but is not necessary to practice the

invention. The processor-based system 700 also includes a read-only memory

(ROM) 710 and may include peripheral devices such as a floppy disk drive 712

15   and a compact disk (CD) ROM drive 714 that also communicates with the CPU

702 over the bus 720 as is well known in the art.

The present invention provides a DRAM having a refresh system

wherein internally generated refresh operations are performed on a periodic basis

20   while avoiding conflicts with controller-generated data accesses.

The present invention has many possible implementations that would

not change the scope of the invention. For example, although the read and/or

write access commands are depicted as being allowed only on every fourth clock cycle, the exact number is, of course, not critical. The number of clock cycles is a function of the time required to perform the refresh operation and the frequency of the DRAM clock. For purposes of describing the present invention, the

5   refresh operation was assumed to require four clock cycles. In addition, it should also be appreciated that the refresh operation may be triggered by any portion of the DRAM clock without deviating from the present invention. Furthermore, the block diagram of Fig. 2 is intended to be a representative hardware configuration for describing the invention; and, therefore, the components used

10   and/or the manner in which they are configured is not critical.

Therefore, while the invention has been described in detail in connection with a preferred embodiment known at the time, it should be readily understood that the invention is not limited to such disclosed embodiment.

15   Rather, the invention can be modified to incorporate any number of variations, alterations, substitutions or equivalent arrangements not heretofore described, but which are commensurate with the spirit and scope of the invention. Accordingly, the invention is not to be seen as limited by the foregoing description, but is only limited by the scope of the appended claims.

What is claimed as new and desired to be protected by Letters Patent of the United States is:

1. A method for refreshing memory cells, comprising:

5   determining that a refresh of said memory cells is required;

determining that a data access is desired; and

initiating said refresh at a predetermined time without delaying said data access.

10   2. The method of claim 1, wherein said act of determining that a data access is desired comprises determining that a data access command has been applied to a command/address bus at a predetermined time slot.

3. The method of claim 1, wherein said act of initiating comprises

15   initiating said refresh within a predetermined time slot of a command/address bus.

4. The method of claim 2 further comprising determining whether said data access command conflicts with said refresh.

20

5. The method of claim 4 further comprising:

determining that said data access command does not conflict with said refresh; and

performing said refresh.

6. The method of claim 4 further comprising:

determining that said data access command does conflict with said

5   refresh;

completing said conflicting data access;

waiting for a next available time slot upon which said refresh may be

initiated; and

initiating said refresh upon reaching said next available time slot

10   without conflicting with a second data access.

7. The method of claim 6 further comprising determining whether a

second data access command has been applied to said command/address bus at

said next available time slot.

15

8. The method of claim 7 further comprising:

determining that a second data access command has been applied to

said command/address bus at said predetermined time slot; and

determining whether said second data access command conflicts with

20   said refresh.

9. The method of claim 8 further comprising:

determining that said second data access command does not conflict with said refresh; and

performing said refresh.

5   10. The method of claim 8 further comprising:

determining that said second data access command does conflict with said refresh;

completing said second conflicting data access;

waiting for a next available time slot upon which said refresh may be

10   initiated; and

initiating said refresh upon reaching said next available time slot without conflicting with a third data access.

11. The method of claim 10 further comprising:

15   satisfying said second data access command by having said data access satisfied with a memory source other than said memory cells such that said memory cells can be refreshed; and

performing said refresh on said memory cells.

20   12. The method of claim 11, wherein said act of satisfying comprises satisfying said second data access command by having said data access satisfied with a static random access memory cache.

13. A system for refreshing memory cells of a dynamic random access memory (DRAM) comprising:

a memory array containing said memory cells;

a communication link for delivering data access commands to said

5   DRAM; and

a controller for operating said memory array in accordance with said data access commands, wherein

said controller is configured to determine that said memory cells require a refresh, and wherein

10   said controller is configured to initiate said refresh at a predetermined time and without delaying a data access triggered by a data access command.

14. The system of claim 13, wherein said communication link comprises a command/address bus.

15

15. The system of claim 13 further comprising a memory source other than said memory cells such that said controller operates said DRAM such that said conflicting data access command is satisfied with said memory source other than said memory cells and wherein said memory cells may be refreshed.

20

16. The system of claim 15, wherein said memory source other than said memory cells comprises a static random access memory (SRAM) cache.

17. The system of claim 13, wherein said controller comprises a refresh controller.

18. The system of claim 13 further comprising:

5   a counter for counting clock cycles of a dynamic random access memory (DRAM) input clock, wherein at least some of said clock cycles define said predetermined time.

19. The system of claim 13, wherein said predetermined time is

10   defined by receiving a predetermined edge of a data access command.

20. The system of claim 19, wherein said predetermined edge is defined as every fourth positive edge of a dynamic random access memory (DRAM) input clock.

15

21. The system of claim 20, wherein said DRAM input clock has a frequency of approximately 300 MHz.

22. The system of claim 13, wherein said communication link

20   comprises a link for communicating read and/or write commands to said memory array.

23. A memory device, comprising:

a memory controller configured to operate said memory device to:

determine that a refresh of said memory cells is required;

determine that a data access is desired; and

initiate said refresh at a predetermined time without delaying said data

5   access.

24. The memory device of claim 23, wherein said memory controller
operates said memory device to determine that a data access command has been
applied to a command/address bus at a predetermined time slot.

10

25. The memory device of claim 23, wherein said memory controller
operates said memory device to initiate said refresh within a predetermined time
slot of a command/address bus.

15   26. The memory device of claim 24, wherein said memory controller
further operates said memory device to determine whether said data access
command conflicts with said refresh.

27. The memory device of claim 26, wherein said memory controller
20   further operates said memory device to:

determine that said data access command does not conflict with said

refresh; and

perform said refresh.

28. The memory device of claim 26, wherein said memory controller further operates said memory device to:

determine that said data access command does conflict with said

5   refresh;

complete said conflicting data access;

wait for a next available time slot upon which said refresh may be initiated; and

initiate said refresh upon reaching said next available time slot without

10   conflicting with a second data access.

29. The memory device of claim 28, wherein said memory controller further operates said memory device to determine whether a second data access command has been applied to said command/address bus at said next available

15   time slot.

30. The memory device of claim 29, wherein said memory controller further operates said memory device to:

determine that a second data access command has been applied to said

20   command/address bus at said predetermined time slot; and

determine whether said second data access command conflicts with said refresh.

31. The memory device of claim 30, wherein said memory controller further operates said memory device to:

determine that said second data access command does not conflict with said refresh; and

5   perform said refresh.

32. The memory device of claim 30, wherein said memory controller further operates said memory device to:

determine that said second data access command does conflict with

10   said refresh;

complete said second conflicting data access;

wait for a next available time slot upon which said refresh may be initiated; and

initiate said refresh upon reaching said next available time slot without

15   conflicting with a third data access.

33. The memory device of claim 32, wherein said memory controller further operates said memory device to:

satisfy said second data access command by having said data access

20   satisfied with a memory source other than said memory cells such that said memory cells can be refreshed; and

perform said refresh on said memory cells.

34. The memory device of claim 33, wherein said memory controller operates said memory device such that said act of satisfying comprises satisfying said second data access command by having said data access satisfied with a static random access memory cache.

5

35. An integrated circuit semiconductor device containing a system for refreshing memory cells of a dynamic random access memory (DRAM), said integrated circuit semiconductor device comprising:

a memory array containing said memory cells;

10   a communication link for delivering data access commands to said DRAM; and

a controller for operating said memory array in accordance with said data access commands, wherein

said controller is configured to determine that said memory cells

15 require a refresh, and wherein

said controller is configured to initiate said refresh at a predetermined time and without delaying a data access triggered by a data access command.

36. The integrated circuit device of claim 35, wherein said

20 communication link comprises a command/address bus.

37. The integrated circuit device of claim 35, wherein said system further comprises a memory source other than said memory cells such that upon

said controller determining that data stored in at least some of said memory cells

is in danger of being lost as a result of not being refreshed, said controller

operates said DRAM such that said conflicting data access command is satisfied

with said memory source other than said memory cells and wherein said memory

5   cells may be refreshed.

38. The integrated circuit device of claim 37, wherein said memory

source other than said memory cells comprises a static random access memory

(SRAM) cache.

10

39. The integrated circuit device of claim 35, wherein said controller

comprises a refresh controller.

40. The integrated circuit device of claim 35, wherein said system

15   further comprises:

a counter for counting clock cycles of a dynamic random access

memory (DRAM) input clock, wherein at least some of said clock cycles

correspond to said predetermined time.

20   41. The integrated circuit device of claim 35, wherein said

predetermined time is defined by receiving a predetermined edge of a data access

command.

42. The integrated circuit device of claim 41, wherein said predetermined edge is defined as every fourth positive edge of a dynamic random access memory (DRAM) input clock.

5   43. The integrated circuit device of claim 42, wherein said dynamic random access memory (DRAM) input clock has a frequency of approximately 300 MHz.

44. The integrated circuit device of claim 35, wherein said

10   communication link comprises a link for communicating read and/or write commands to said memory array.

45. A processor-based system, comprising:

a processor; and

15   a dynamic random access memory (DRAM) coupled to said processor, said dynamic random access memory having a system for refreshing memory cells in said dynamic random access memory, said system comprising:

a memory array containing said memory cells;

a communication link for delivering data access commands to said

20   DRAM; and

a controller for operating said memory array in accordance with said data access commands, wherein

said controller is configured to determine that said memory cells require a refresh, and wherein

said controller is configured to initiate said refresh at a predetermined time and without delaying a data access triggered by a data

5   access command.

46. The processor-based system of claim 45, wherein said communication link comprises a command/address bus.

10   47. The processor-based system of claim 45, wherein said system for refreshing memory cells further comprises a memory source other than said memory cells such that said controller operates said DRAM such that said conflicting data access command is satisfied with said memory source other than said memory cells and wherein said memory cells may be refreshed.

15

48. The processor-based system of claim 47, wherein said memory source other than said memory cells comprises a static random access memory (SRAM) cache.

20   49. The processor-based system of claim 45, wherein said controller comprises a refresh controller.

50. The processor-based system of claim 45, wherein said system for refreshing memory cells further comprises:

a counter for counting clock cycles of a dynamic random access memory (DRAM) input clock, wherein at least some of said clock cycles define

5   said predetermined time.

51. The processor-based system of claim 45, wherein said predetermined time is defined by receiving a predetermined edge of a data access command.

10

52. The processor-based system of claim 45, wherein said predetermined edge is defined as every fourth positive edge of a dynamic random access memory (DRAM) input clock.

15   53. The processor-based system of claim 52, wherein said DRAM input clock has a frequency of approximately 300 MHz.

54. The processor-based system of claim 45, wherein said communication link comprises a link for communicating read and/or write

20   commands to said memory array.

ABSTRACT

A synchronous DRAM is provided having specified time slots (e.g.,
every multiple of 4 clock pulses of a DRAM input clock) within which read or
write commands may be entered on the command/address bus. During

5   operation, the DRAM performs internally generated refresh operations on a
periodic basis while avoiding collisions with controller-generated data accesses.
An internal refresh cycle can be executed without interfering with any data
accesses by starting the refresh after decoding a non-conflicting command in one
of these time slots and finishing before the next command time slot. If an

10   internal refresh operation is delayed (e.g., by the decoding of a conflicting access
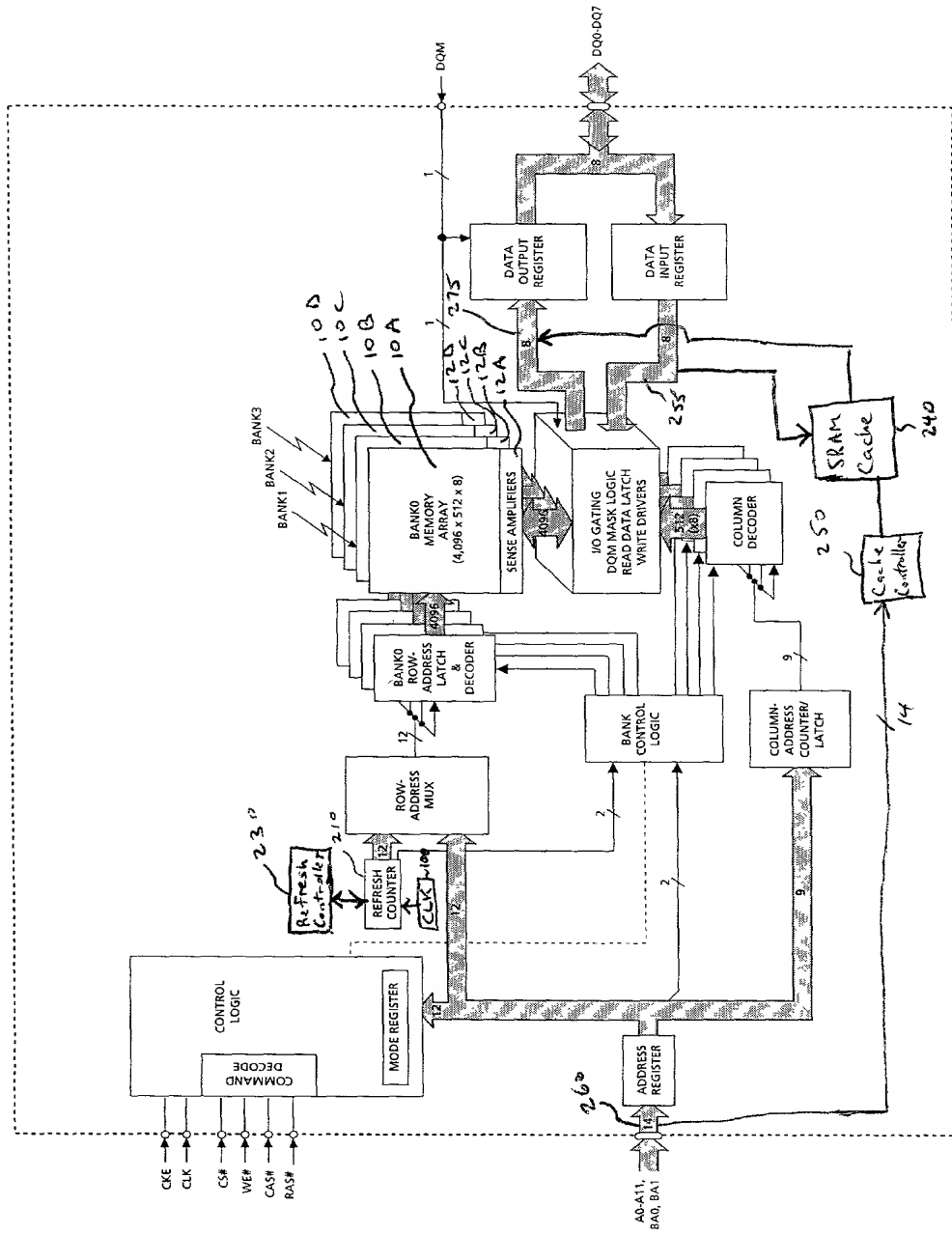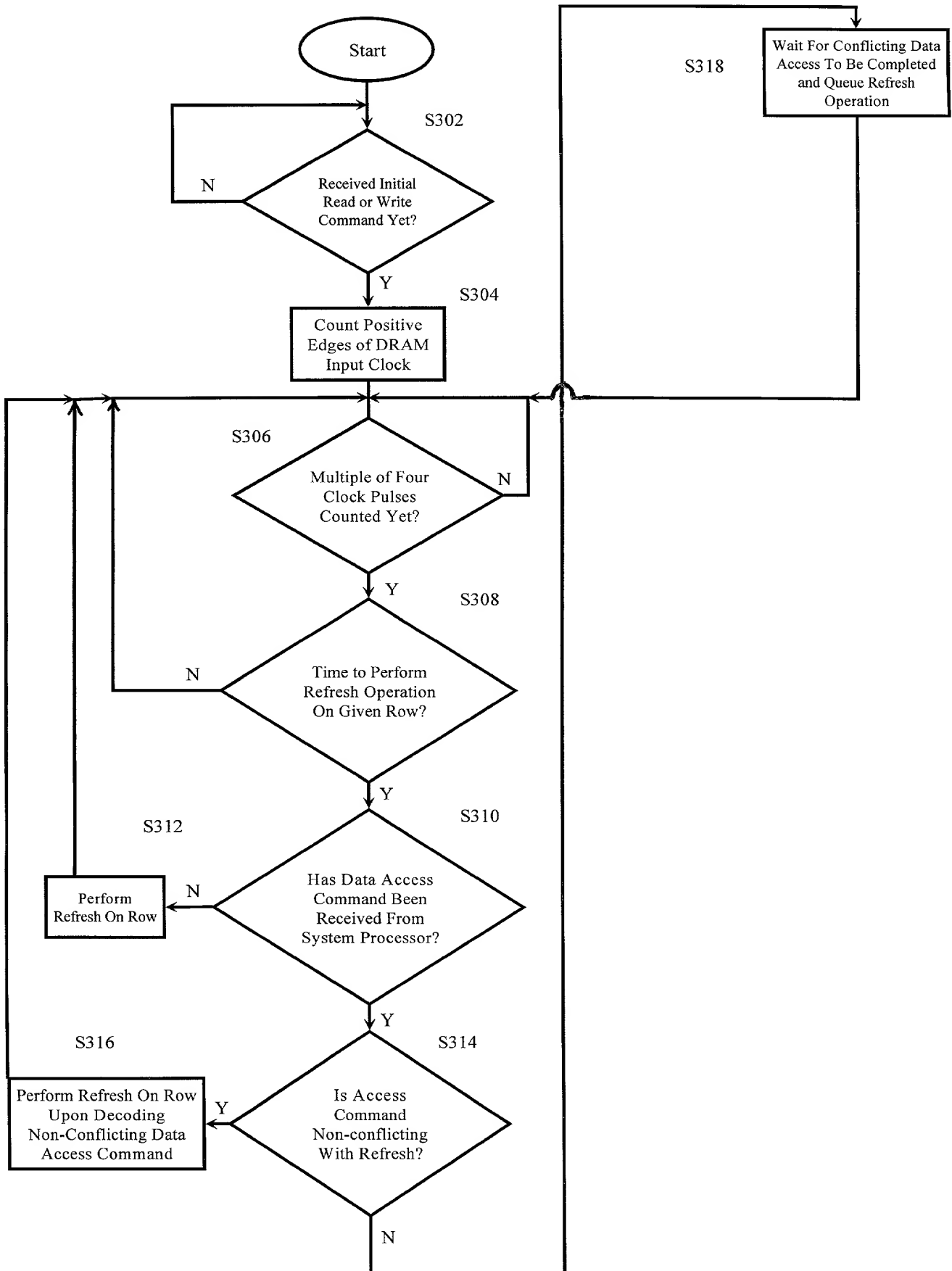command) it will be completed at the earliest opportunity thereafter.

Fig. 1

Fig. 2

CONTROL LOGIC

COMMAND DECODE

MODE REGISTER

CKE
CLK
CS#
WE#
CASH
RASH

A0-A11, BA0, BA1

ADDRESS REGISTER

REFRESH COUNTER

Refresh Controller

CLK

ROW-ADDRESS MUX

BANK0 ROW-ADDRESS LATCH & DECODER

BANK CONTROL LOGIC

COLUMN-ADDRESS COUNTER/LATCH

BANK3
BANK2
BANK1
BANK0 MEMORY ARRAY (4,096 x 512 x 8)

SENSE AMPLIFIERS

I/O GATING
DQM MASK LOGIC
READ DATA LATCH
WRITE DRIVERS

COLUMN DECODER

SRAM Cache

Cache Controller

DATA OUTPUT REGISTER

DATA INPUT REGISTER

DQM

DQ0-DQ7

10D 10C 10B 10A
12D 12C 12B 12A
275
255
240
250
210
230
260

Fig. 3   S300

```
                              ┌──────────┐
                              │   Start   │
                              └──────────┘
                                    │
                                    ▼              S302
                        ◇─────────────────────◇
              N         │  Received Initial    │
      ┌───────────────  │  Read or Write       │
      │                 │  Command Yet?        │
      │                 ◇─────────────────────◇
      │                           │ Y
      │                           ▼           S304
      │                 ┌──────────────────┐
      │                 │  Count Positive  │
      │                 │  Edges of DRAM   │
      │                 │  Input Clock     │
      │                 └──────────────────┘
```

S318 — Wait For Conflicting Data Access To Be Completed and Queue Refresh Operation

S306 — Multiple of Four Clock Pulses Counted Yet?   N

S308 — Time to Perform Refresh Operation On Given Row?   N

S310 — Has Data Access Command Been Received From System Processor?   N

S312 — Perform Refresh On Row

S314 — Is Access Command Non-conflicting With Refresh?   N

S316 — Perform Refresh On Row Upon Decoding Non-Conflicting Data Access Command   Y
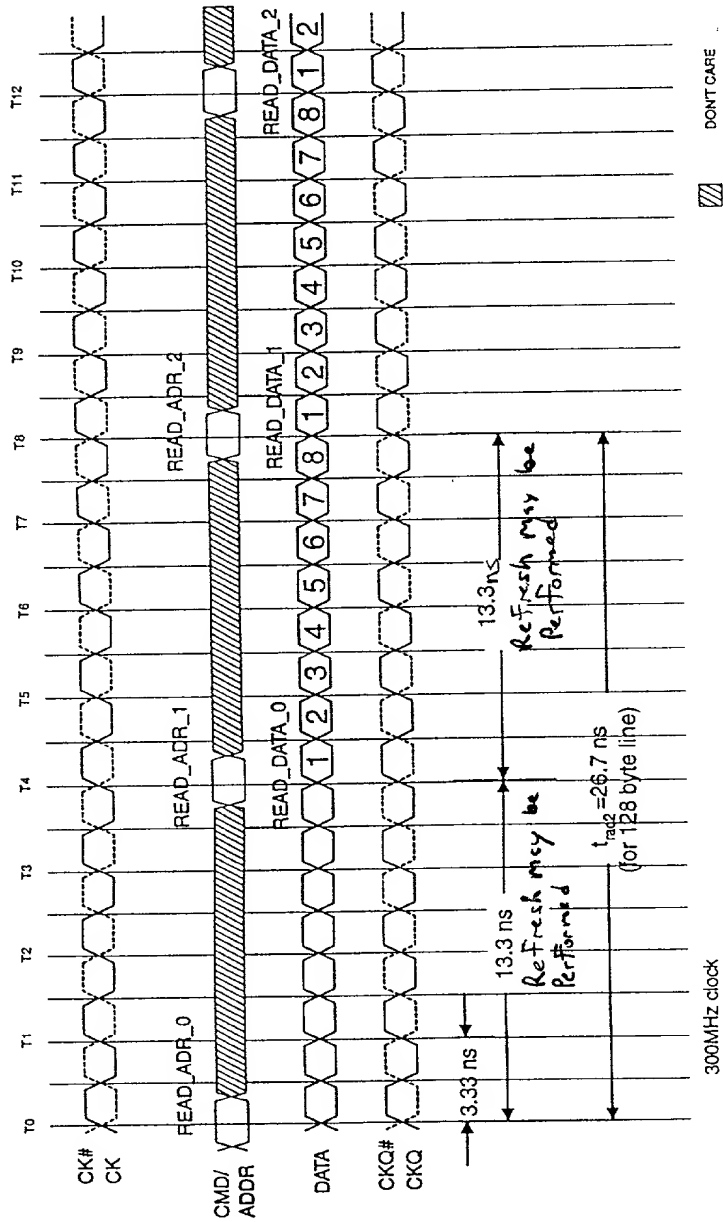
READ - READ Timing Diagram

Fig. 4

WRITE - WRITE Timing Diagram

Fig. 5

READ - WRITE - READ Timing Diagram

300MHz clock

Fig. 6

700

702

CPU

706

I/O
DEVICE

708

DRAM

716

712

FLOPPY
DISK
DRIVE

710

ROM

718

714

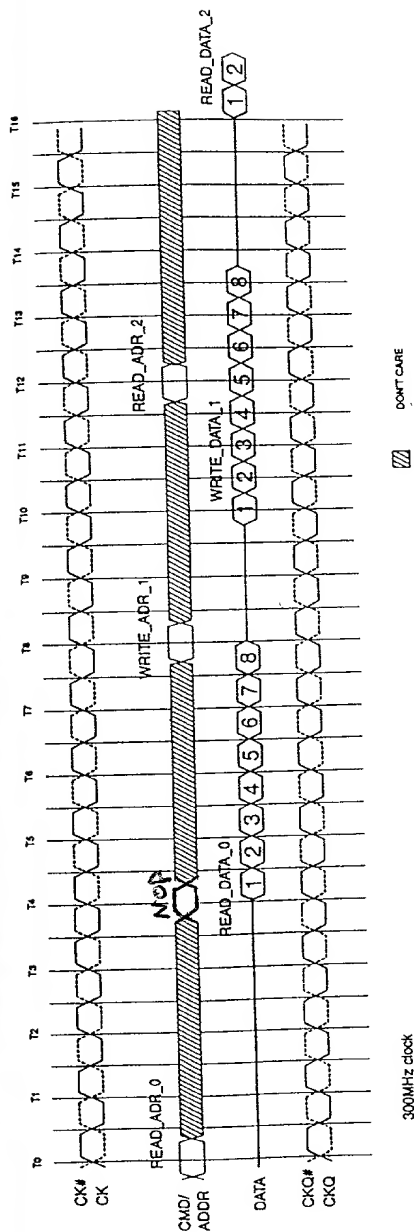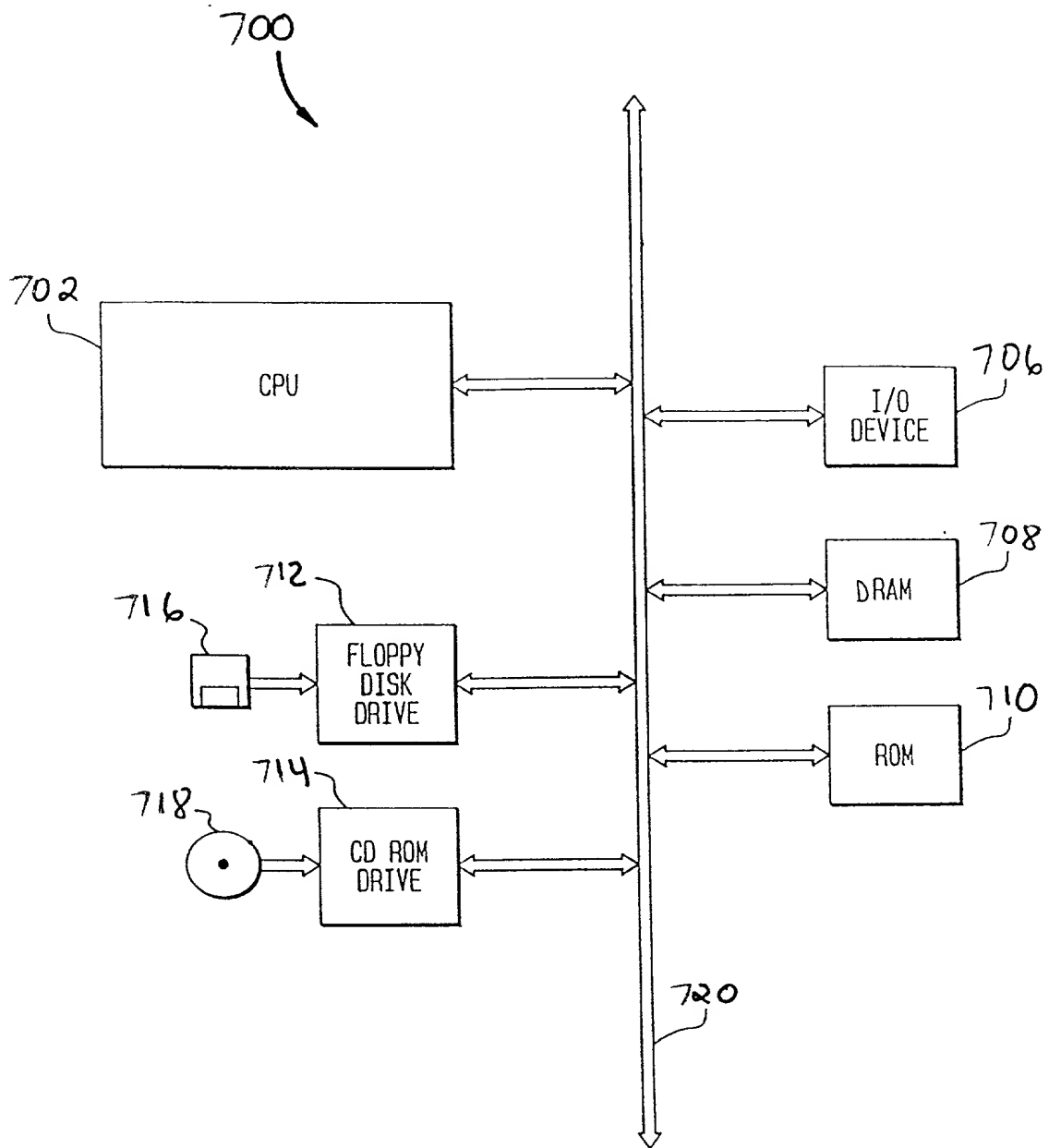CD ROM
DRIVE

720
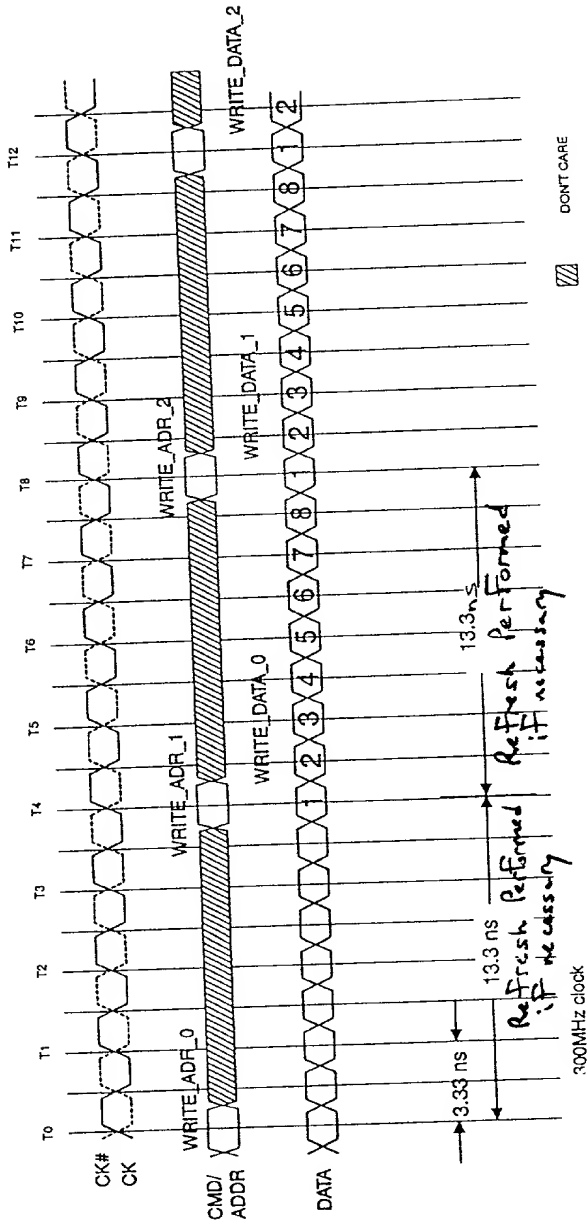
Fig. 7

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

DECLARATION FOR PATENT APPLICATION

As the below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name.

I believe I am the original, first and sole inventor of the subject matter which is claimed and for which a patent is sought on the invention entitled:

DRAM WITH HIDDEN REFRESH.

The specification of which is attached hereto.

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by an amendment, if any, specifically referred to in this oath or declaration.

I acknowledge the duty to disclose all information known to me which is material to patentability as defined in Title 37, Code of Federal Regulations, § 1.56.

I hereby claim foreign priority benefits under Title 35, United States Code, § 119/365 of any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on which priority is claimed:

None

I hereby claim the benefit under Title 35, United States Code, § 120/365 of any United States and PCT international application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, § 112, I acknowledge the duty to disclose all information known to me to be material to patentability as defined in Title 37, Code of Federal Regulations, § 1.56(a) which became available between the filing date of the prior application and the national or PCT international filing date of this application:

None

I agree that, when requested, I shall, without charge to Micron Technology, Inc. but at its expense, sign all papers, and do all acts which may be necessary, desirable or convenient in connection with said applications, patents, or other forms of protection.
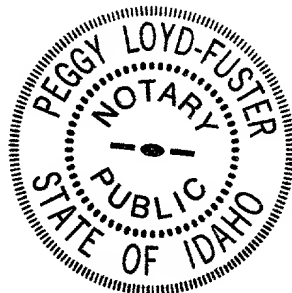
Kevin J. Ryan

Date: 8/16/00

United States of America)
State of Idaho)ss.:
County of Ada)

On this 16th day of August , 2000, before me personally came Kevin J. Ryan, to me known to be the individual described in and who executed the foregoing instrument, and acknowledged execution of the same.

Notary Public

1164282 v1; _YD601! DOC

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of:
Kevin J. Ryan

Serial No.: Not yet Assigned   Group Art Unit: Not yet assigned

Filed: Herewith   Examiner: Not yet Assigned

For: DRAM WITH HIDDEN REFRESH

Assistant Commissioner for Patents
Washington, D.C. 20231

POWER OF ATTORNEY BY ASSIGNEE AND
CERTIFICATE BY ASSIGNEE UNDER 37 CFR § 3.73(b)

Micron Technology, Inc., assignee of the entire right, title and interest in the above-identified application by virtue of the assignment attached hereto (which is also being submitted concurrently for recordation), hereby appoints the attorneys and agents of the firm of Dickstein Shapiro Morin & Oshinsky LLP located at 2101 L Street, NW, Washington, DC 20037-1526, listed as follows: Gary M. Hoffman, 26,411; Thomas J. D'Amico, 28,371; Donald A. Gregory, 28,954; James W. Brady, Jr., 32,115; Jon D. Grossman, 32,699; Mark J. Thronson, 33,082; Laurence D. Fisher, 37,131; Brian A. Lemm, 43,748; Gianni Minutoli, 41,198; Eric Oliver, 35,307; William E. Powell, III, 39,803; Edwin Oh, 45,319; and Salvatore P. Tamburo, 45,153; and also attorneys Michael L. Lynch, 30,871; Lia M. Pappas, 34,095; W. Eric Webostad, 35,406; and Charles B. Brantley, II, 38,086 of Micron Technology, Inc. as its attorneys with full power of substitution to prosecute this application and to transact all business in the Patent and Trademark Office in connection therewith.

The assignee certifies that the above-identified assignment has been reviewed and to the best of the assignee's knowledge and belief, title is in the assignee.

Please direct all correspondence regarding this application to the following:

Thomas J. D'Amico, Esq.
Dickstein Shapiro Morin & Oshinsky LLP
2101 L Street, NW
Washington, D.C. 20037-1526
Telephone: (202) 828-2232
Facsimile: (202) 887-0689

MICRON TECHNOLOGY, INC.

Michael L. Lynch
Chief Patent Counsel
Registration No. 30,871

Dated: _8-16-ω_

2